AI/ML Challenge Tech Talk

Ryan Day

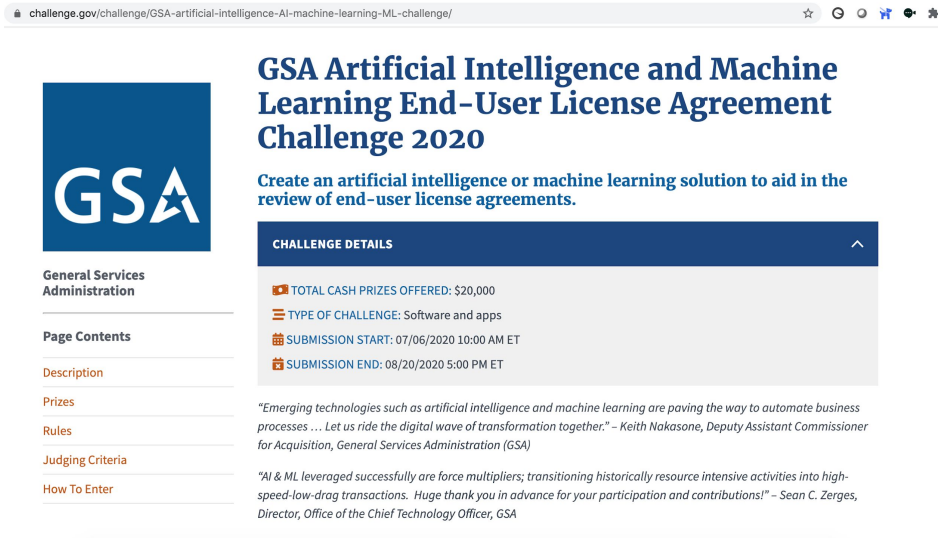
December 2020

CHIEF TECHNOLOGY OFFICER

Today I'll be sharing some of the technical and data science information that we learned from our recent AI/ML challenge.

The OCTO works to promote emerging technology and spread understanding of technology in the agency.

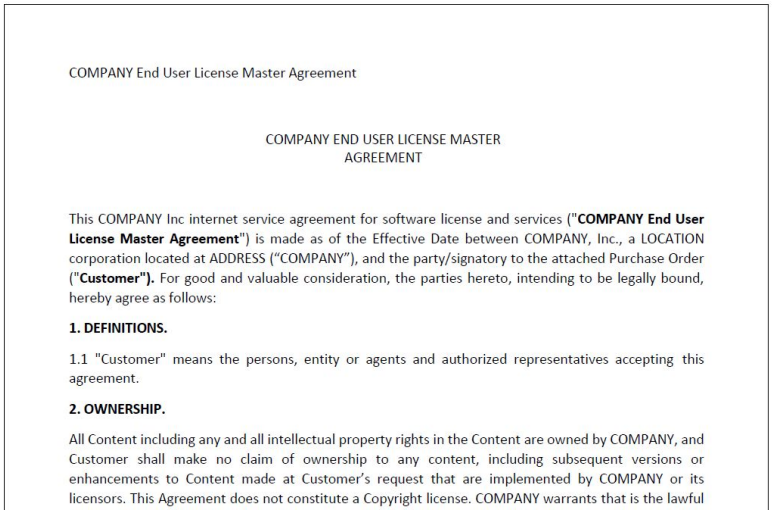
Our Challenge Event



During the summer of 2020, the OCTO and FAS ITC collaborated to host an online machine learning challenge.

Full details are at this website: https://www.challenge.gov/challenge/GSA-artificial-intelligence-AI-machine-learning-ML-challenge/

What is an End User License Agreement (EULA)?



COMPANY End User License Master Agreement

COMPANY END USER LICENSE MASTER AGREEMENT

This COMPANY Inc internet service agreement for software license and services ("**COMPANY End User License Master Agreement**") is made as of the Effective Date between COMPANY, Inc., a LOCATION corporation located at ADDRESS ("COMPANY"), and the party/signatory to the attached Purchase Order ("**Customer**"). For good and valuable consideration, the parties hereto, intending to be legally bound, hereby agree as follows:

1. DEFINITIONS.

1.1 "Customer" means the persons, entity or agents and authorized representatives accepting this agreement.

2. OWNERSHIP.

All Content including any and all intellectual property rights in the Content are owned by COMPANY, and Customer shall make no claim of ownership to any content, including subsequent versions or enhancements to Content made at Customer's request that are implemented by COMPANY or its licensors. This Agreement does not constitute a Copyright license. COMPANY warrants that is the lawful

CHIEF TECHNOLOGY OFFICER

Let's start with the business problem wanted to solve:

The goal of this challenge is to develop an artificial intelligence (AI), or machine learning (ML) solution that will review end-user license agreements (EULA) for terms and conditions that are unacceptable to the government. On average it takes all parties involved approximately 7-14 days to review an EULA and ensure that all unacceptable terms and conditions are identified.

A EULA details the rights and restrictions which apply to the use of software or services. It can also be known as a software license agreement or acceptable use policy. As part of the acquisition process of software or services, GSA reviews the associated EULAs. This review must be completed prior to awarding new contracts or modifying existing contracts. A GSA contracting officer (CO) reviews applicable EULAs for terms and conditions that are not in accordance with Federal law and regulations. The CO may also coordinate a legal review with the Office of General Counsel if they feel it is warranted. Should EULAs contain language that would conflict with Federal law and regulations, the CO must negotiate changes to the EULA to remove the problematic language.

We are looking for a solution that will use AI and/or ML to improve this

manual process. The solution will include a user interface that GSA will use to process the documents and identify unacceptable clauses/terms in the EULAs. Watch our AI / ML challenge video to learn more about the desired solution.

This solution will decrease the time spent manually reviewing EULAs and free resources to focus on other aspects of the acquisition process. It will also improve the accuracy and consistency of the review process.

Why would clauses be unacceptable to government?

Attachment B

Schedule 70 EULA Matrix

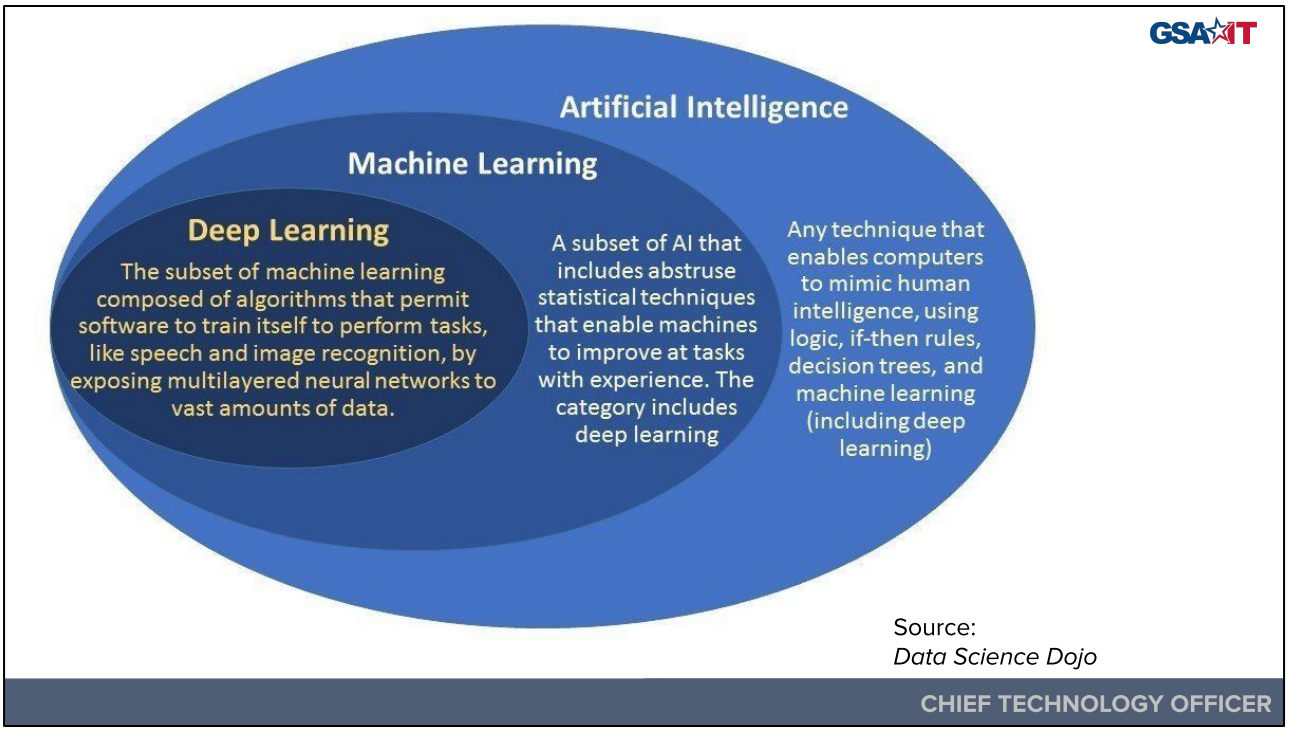
Both GSA and Government ordering activities placing orders under GSA Schedule 70 contracts are required to comply with the FAR clause at 12.212(a), which provides, in relevant part, that commercial computer software and documentation shall be acquired under licenses customarily provided to the public "to the extent such licenses are consistent with Federal law and otherwise satisfy the Government's needs."

Below is a list of terms and conditions commonly occurring in software manufacturers' unmodified commercial agreements that are inconsistent with Federal law and Government needs and, therefore, with FAR 12.212(a). The terms and conditions listed below are non-compliant regardless of the type of agreement: end-user license agreements (EULAs), maintenance agreements, terms of service (TOS), etc.

In order to avoid delays caused by legal review and negotiation of each individual set of terms, or loss of business that occurs when the terms are added on Schedule unmodified and ordering activities later decline to place an order because of non-conforming terms, manufacturers (and/or dealers or resellers, where appropriate) should create compliant agreements that do not contain the clauses listed below.

	Terms and conditions	Problem/recommendation
1	Definition of contracting parties	The Government customer (licensee), under GSA Schedule contracts, is the "ordering activity," defined as an "entity authorized to order under GSA Schedule contracts as defined in GSA Order ADM4800.2G, as may be revised from time to time." The licensee or customer cannot be an individual because any implication of individual licensing triggers the requirement for legal review by Federal employee unions. Conversely, because of competition rules, the contractor must be defined as a single entity even if the contractor is part of a corporate group. The Government cannot contract with the group, or in the alternative with a set of contracting parties.
2	Contract formation via using.	Under FAR 1.601(a), in an acquisition involving the use of

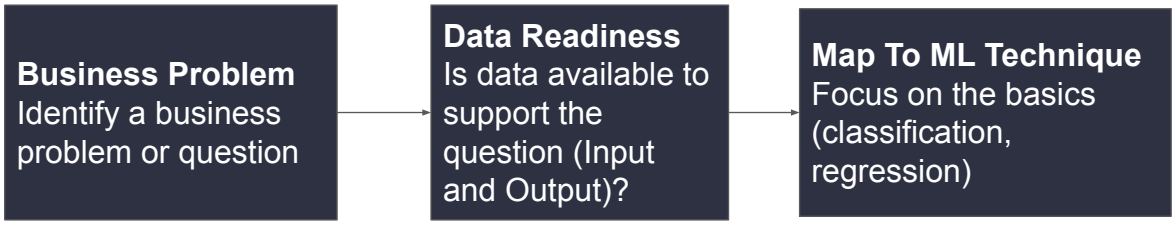
There are a variety of reasons a clause might be unacceptable to government. We provided “Attachment B” to the teams in the reference materials: https://github.com/GSA/ai-ml-challenge-2020/tree/master/reference



This shows how AI & ML relate to each other.

GSA IT

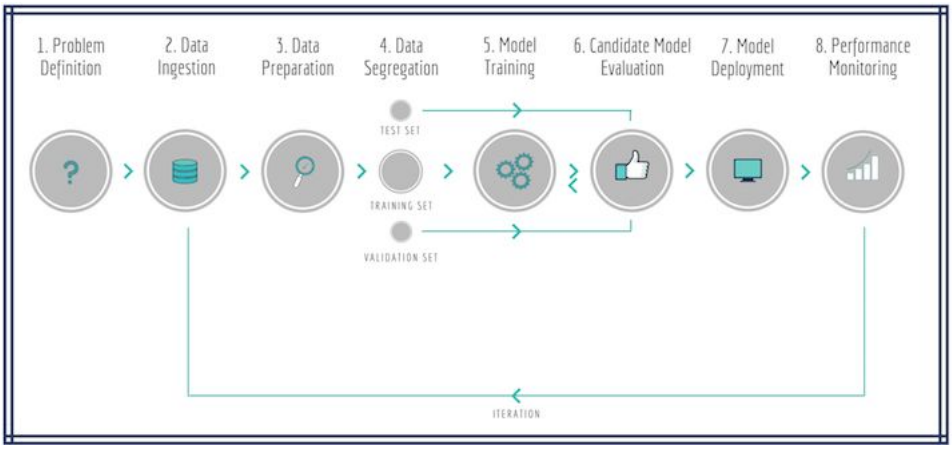
Machine Learning Flow



This is a typical machine learning flow.

GSA IT

Machine Learning Pipeline

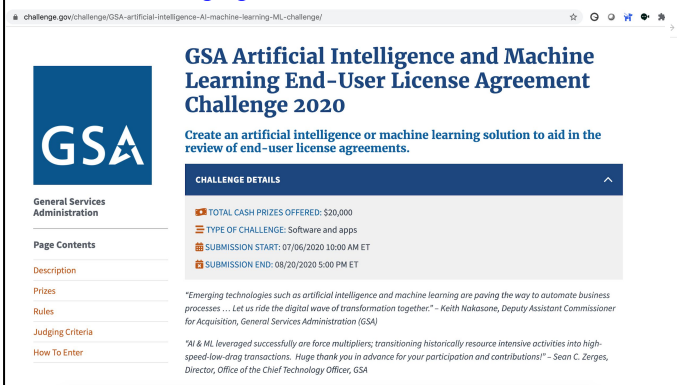


Source: Semi Koen, Towards Data Science: https://towardsdatascience.com/architecting-a-machine-learning-pipeline-a847f094d1c7

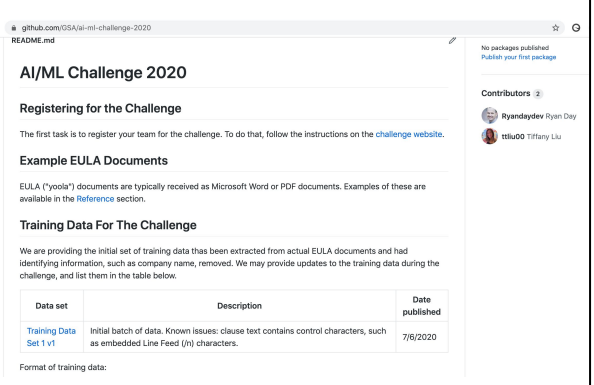
This is a view of a machine learning pipeline.
https://towardsdatascience.com/architecting-a-machine-learning-pipeline-a847f094d1c7

Our Challenge

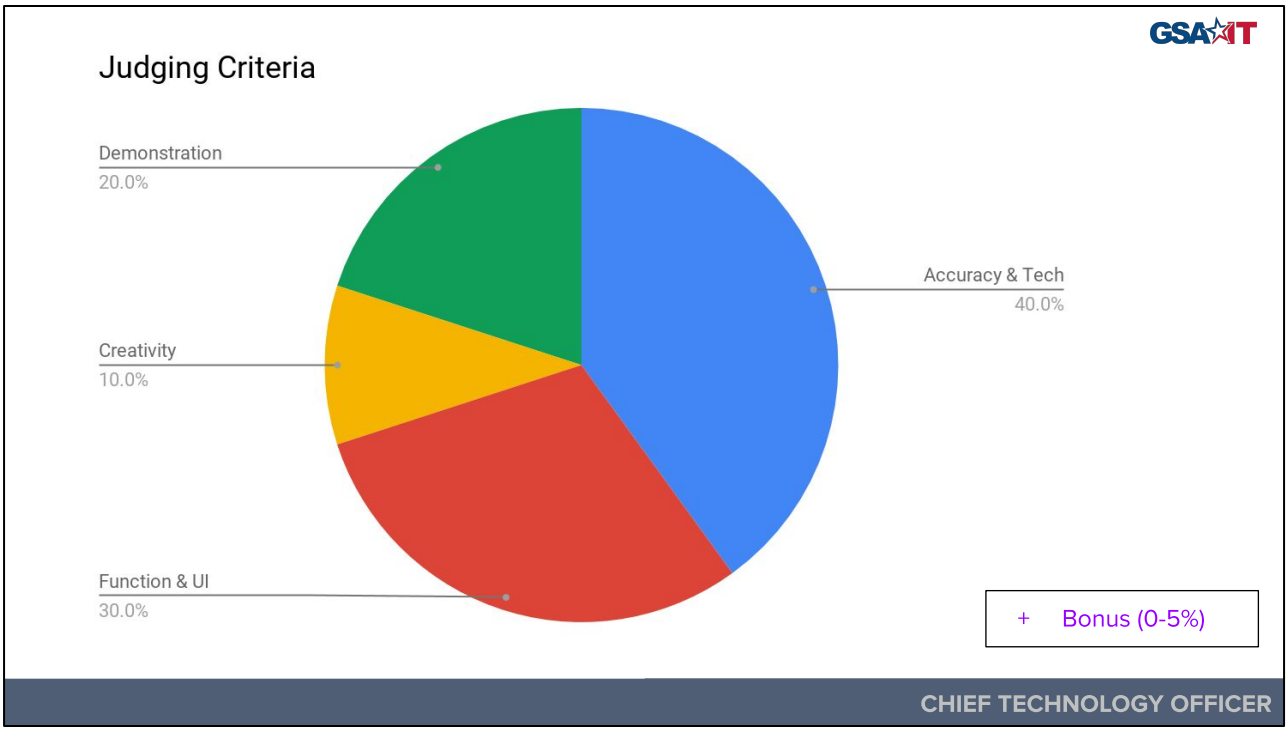
Challenge.gov



Github.com

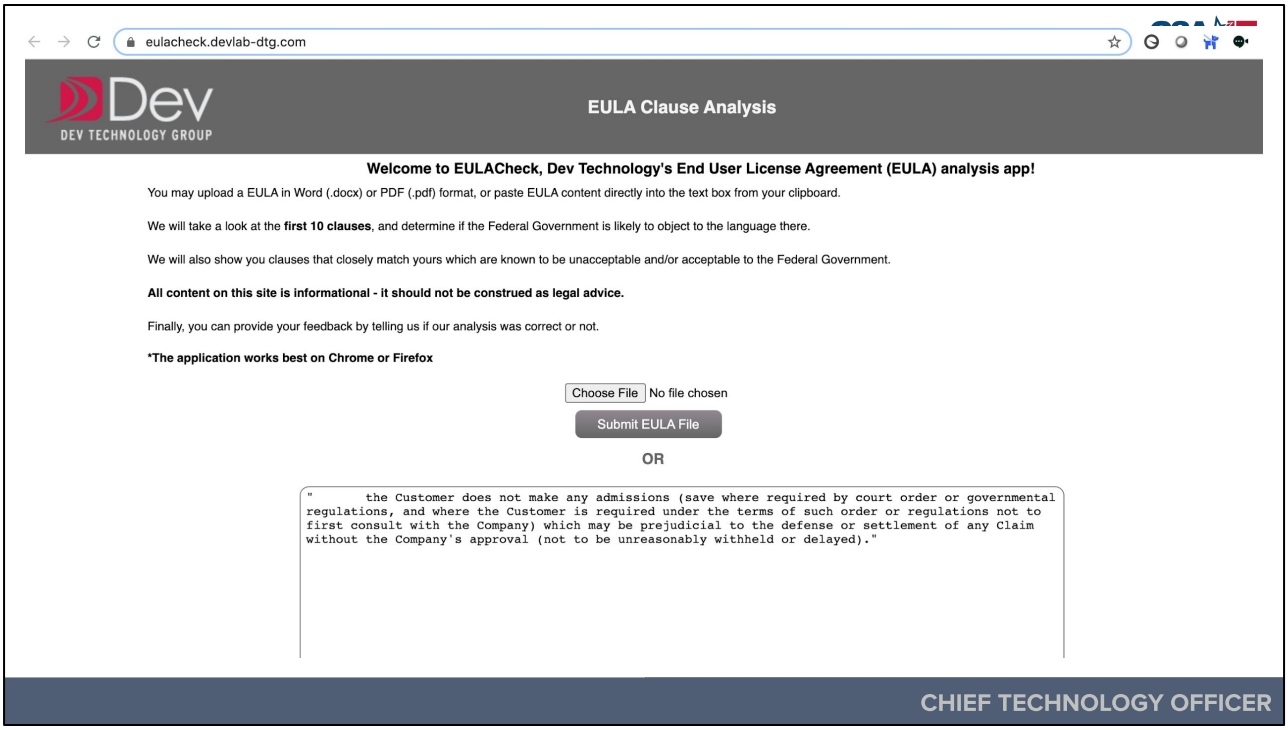


We had instructions on challenge.gov and github.
Here were the github instructions:
https://github.com/GSA/ai-ml-challenge-2020/blob/master/README.md



These were the scoring criteria. We published a scoring rubric to explain them.
https://github.com/GSA/ai-ml-challenge-2020/blob/master/reference/AI_ML%20Challenge%20Scoring%20Rubric.pdf

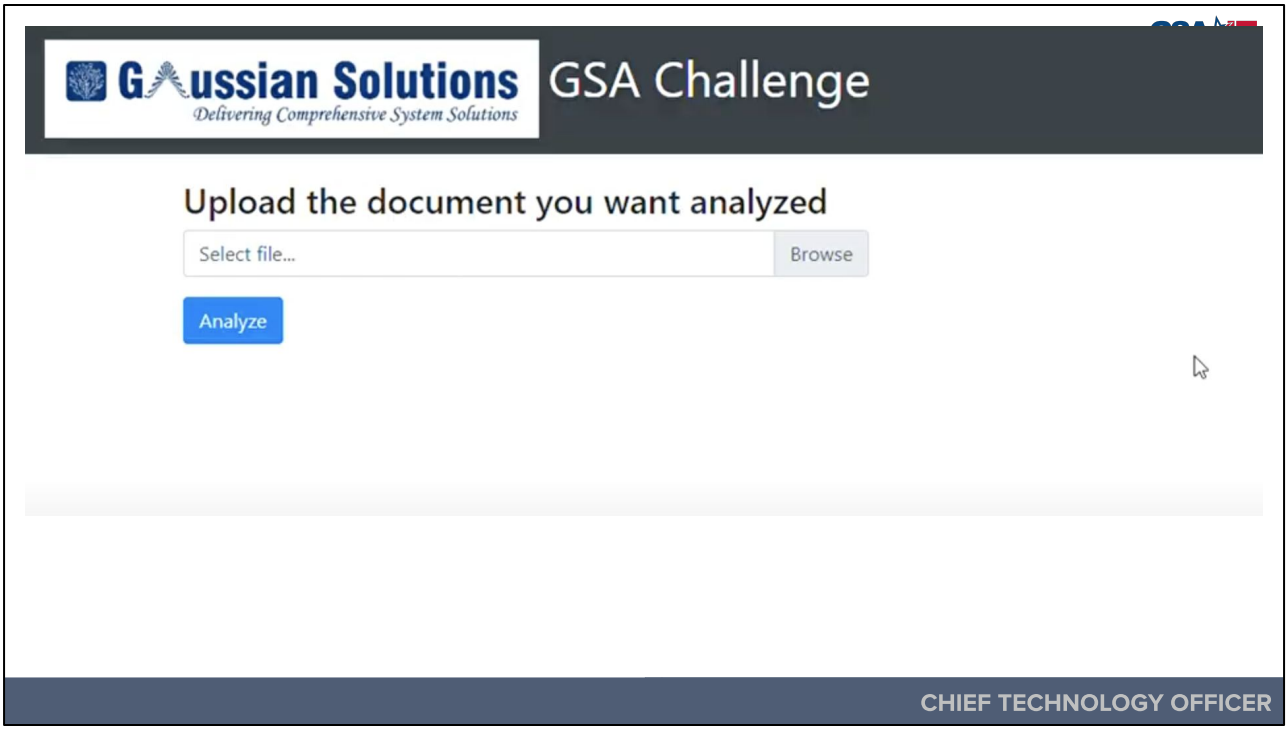
These were the three winners.

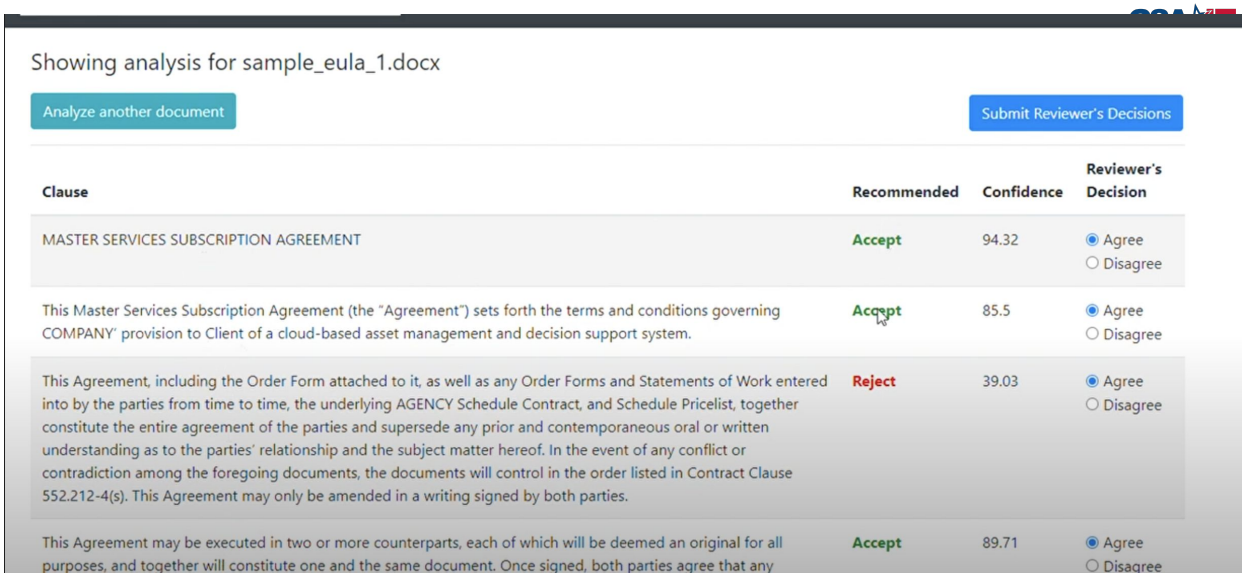

eulacheck.devlab-dtg.com
Dev
DEV TECHNOLOGY GROUP
EULA Clause Analysis
Welcome to EULACheck, Dev Technology's End User License Agreement (EULA) analysis app!
You may upload a EULA in Word (.docx) or PDF (.pdf) format, or paste EULA content directly into the text box from your clipboard.
We will take a look at the first 10 clauses, and determine if the Federal Government is likely to object to the language there.
We will also show you clauses that closely match yours which are known to be unacceptable and/or acceptable to the Federal Government.
All content on this site is informational - it should not be construed as legal advice.
Finally, you can provide your feedback by telling us if our analysis was correct or not.
*The application works best on Chrome or Firefox
Choose File
No file chosen
Submit EULA File
OR
" the Customer does not make any admissions (save where required by court order or governmental regulations, and where the Customer is required under the terms of such order or regulations not to first consult with the Company) which may be prejudicial to the defense or settlement of any Claim without the Company's approval (not to be unreasonably withheld or delayed)."
CHIEF TECHNOLOGY OFFICER

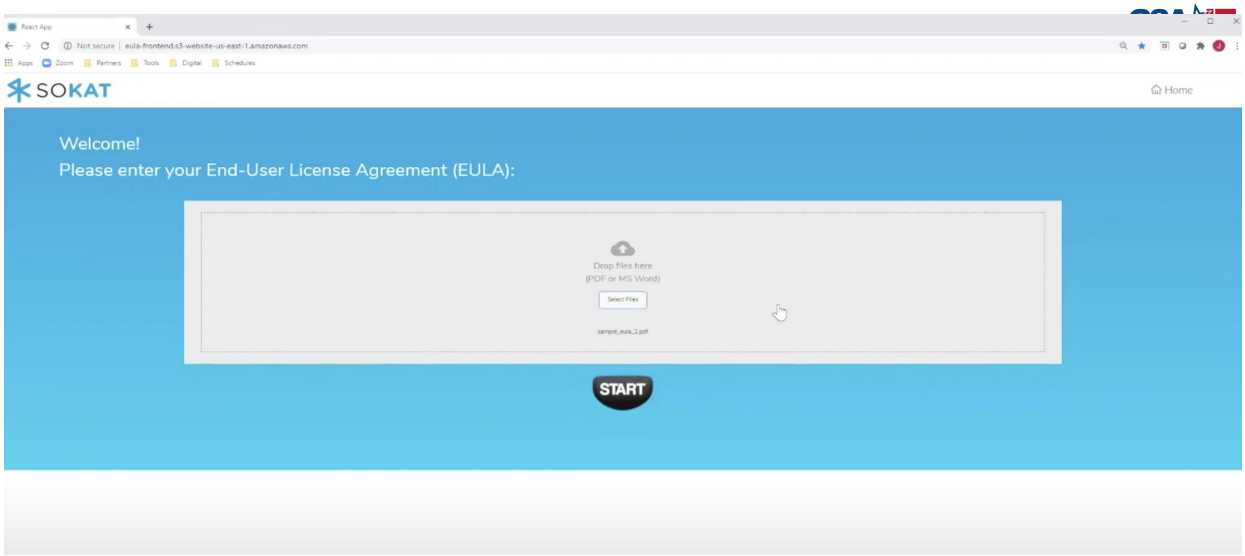
Requests. Company will notify Customer before Customer exceeds the Tile Request Use Limit indicated on the Order Form. If Customer exceeds its Tile Request Use Limits during the License Term, Company will invoice Customer for Overages on written notice (which may be by email). If, after 30 days from the date of that written notice, Customer continues to exceed its Tile Request Use Limit, Company may stop providing the Service to the Customerinitiate a claim with the Contracting Officer under the Contract Disputes Act.

Submit Clause Text

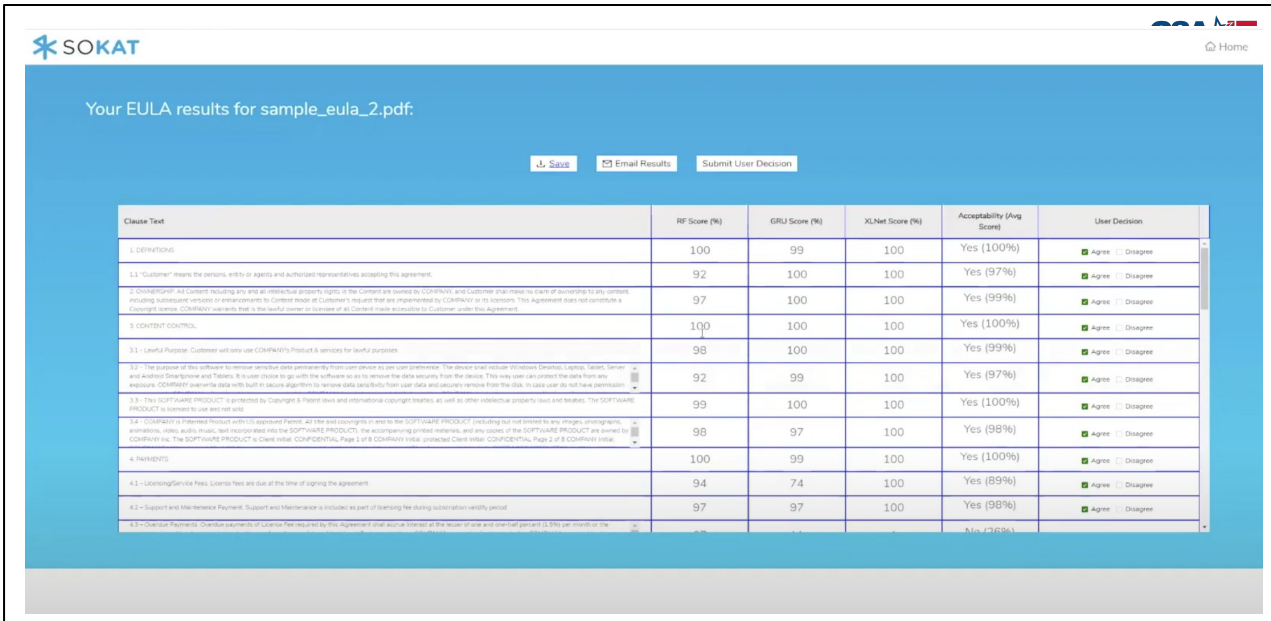
Clause	Acceptable?	Feedback	Closest Acceptable Match	Closest Unacceptable Match
Requests. Company will notify Customer before Customer exceeds the Tile Request Use Limit indicated on the Order Form. If Customer exceeds its Tile Request Use Limits during the License Term, Company will invoice Customer for Overages on written notice (which may be by email). If, after 30 days from the date of that written notice, Customer continues to exceed its Tile Request Use Limit, Company may stop providing the Service to the Customerinitiate a claim with the Contracting Officer under the Contract Disputes Act.	Acceptable 99%	○ Acceptable ○ Unacceptable ○ Not Sure ○ Not a Clause	Term of this Addendum. This Addendum will commence on the Addendum Effective Date and continue for a period of twelve months ("Initial Addendum Term"). Upon the effective date of termination of this Addendum in accordance with the Contract Disputes Act, Client's access to the Hosted Service provided pursuant to this Addendum (and all licenses granted under this Addendum) will cease and COMPANY will delete all backed-up Client Data from the Hosting Infrastructure within 30 days of termination of this Agreement.	Company warrants that the Service will, for a period of sixty (60) days from the date of your receipt, perform substantially in accordance with Service written materials accompanying it


Gaussian Solutions
Delivering Comprehensive System Solutions
GSA Challenge
Upload the document you want analyzed
Select file...
Browse
Analyze
CHIEF TECHNOLOGY OFFICER

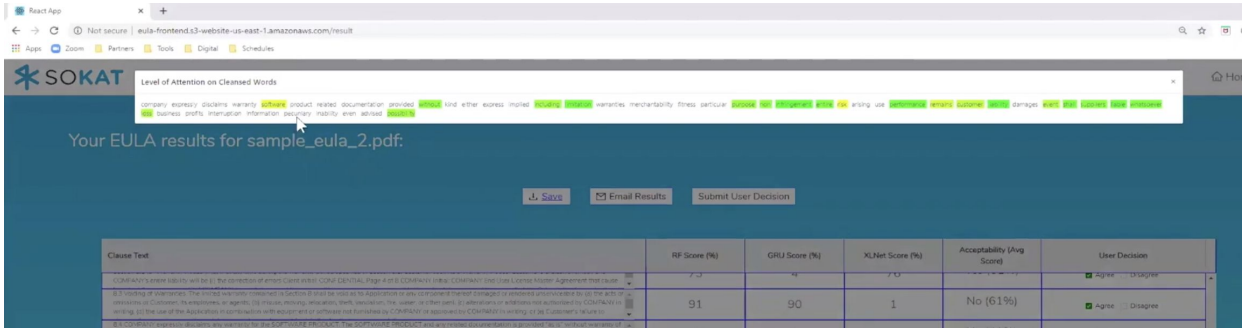

Showing analysis for sample_eula_1.docx
Analyze another document
Submit Reviewer's Decisions
Clause
Recommended
Confidence
Reviewer's Decision
MASTER SERVICES SUBSCRIPTION AGREEMENT
Accept
94.32
Agree
Disagree
This Master Services Subscription Agreement (the "Agreement") sets forth the terms and conditions governing COMPANY' provision to Client of a cloud-based asset management and decision support system.
Accept
85.5
Agree
Disagree
This Agreement, including the Order Form attached to it, as well as any Order Forms and Statements of Work entered into by the parties from time to time, the underlying AGENCY Schedule Contract, and Schedule Pricelist, together constitute the entire agreement of the parties and supersede any prior and contemporaneous oral or written understanding as to the parties' relationship and the subject matter hereof. In the event of any conflict or contradiction among the foregoing documents, the documents will control in the order listed in Contract Clause 552.212-4(s). This Agreement may only be amended in a writing signed by both parties.
Reject
39.03
Agree
Disagree
This Agreement may be executed in two or more counterparts, each of which will be deemed an original for all purposes, and together will constitute one and the same document. Once signed, both parties agree that any
Accept
89.71
Agree
Disagree



CHIEF TECHNOLOGY OFFICER



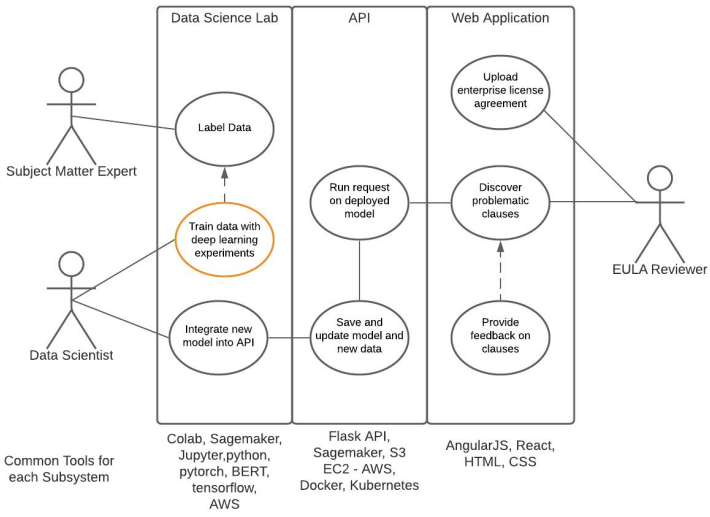
CHIEF TECHNOLOGY OFFICER


React App
Not secure | eula-frontend.s3-website-us-east-1.amazonaws.com/result
Apps
Zoom
Partners
Tools
Digital
Schedules
SOKAT
Level of Attention on Cleansed Words
Your EULA results for sample_eula_2.pdf:
Save
Email Results
Submit User Decision
Clause Text
RF Score (%)
GRU Score (%)
XLNet Score (%)
Acceptability (Avg Score)
User Decision
91
90
1
No (61%)
Agree
Disagree

How Do These Apps Work?

GSA IT

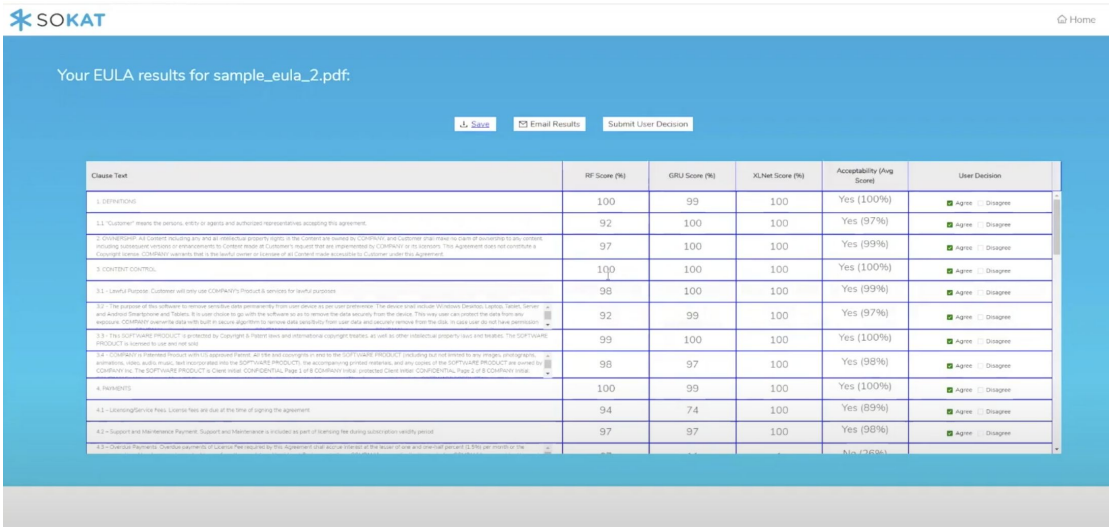
Data Science Deployment Architecture



We found a lot of similarities between the solutions we received.

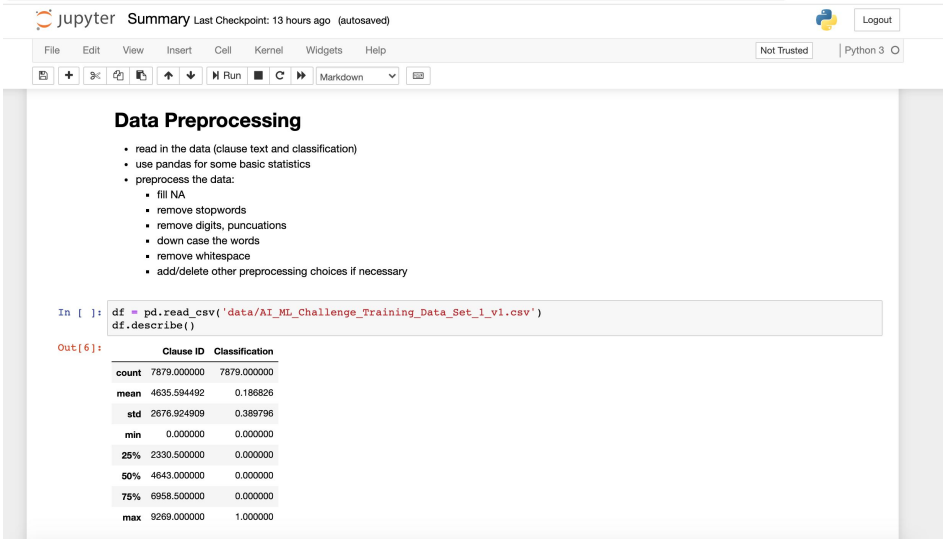
This is a big-picture view of the typical architecture that teams used.

How Did The Teams Work?



I am going to demonstrate the process followed by one of the teams.

Data Science Tool - Jupyter Notebook (Python)



	Clause ID	Classification
count	7879.000000	7879.000000
mean	4635.594492	0.186826
std	2676.924909	0.389796
min	0.000000	0.000000
25%	2330.500000	0.000000
50%	4643.000000	0.000000
75%	6958.500000	0.000000
max	9269.000000	1.000000

The team used Jupyter Notebook and Python for their data science work.

Finished Application Architecture

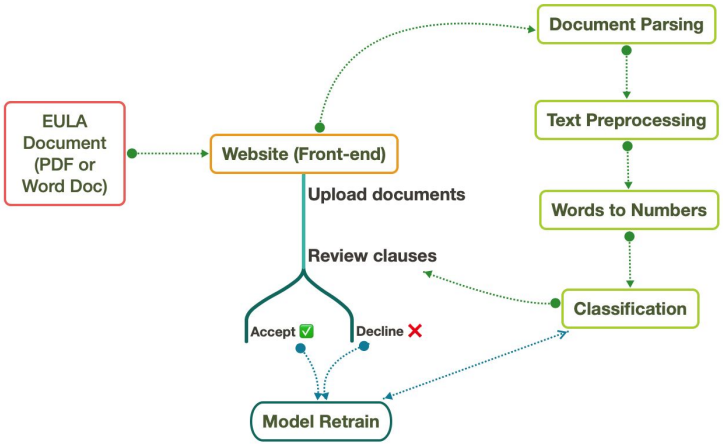
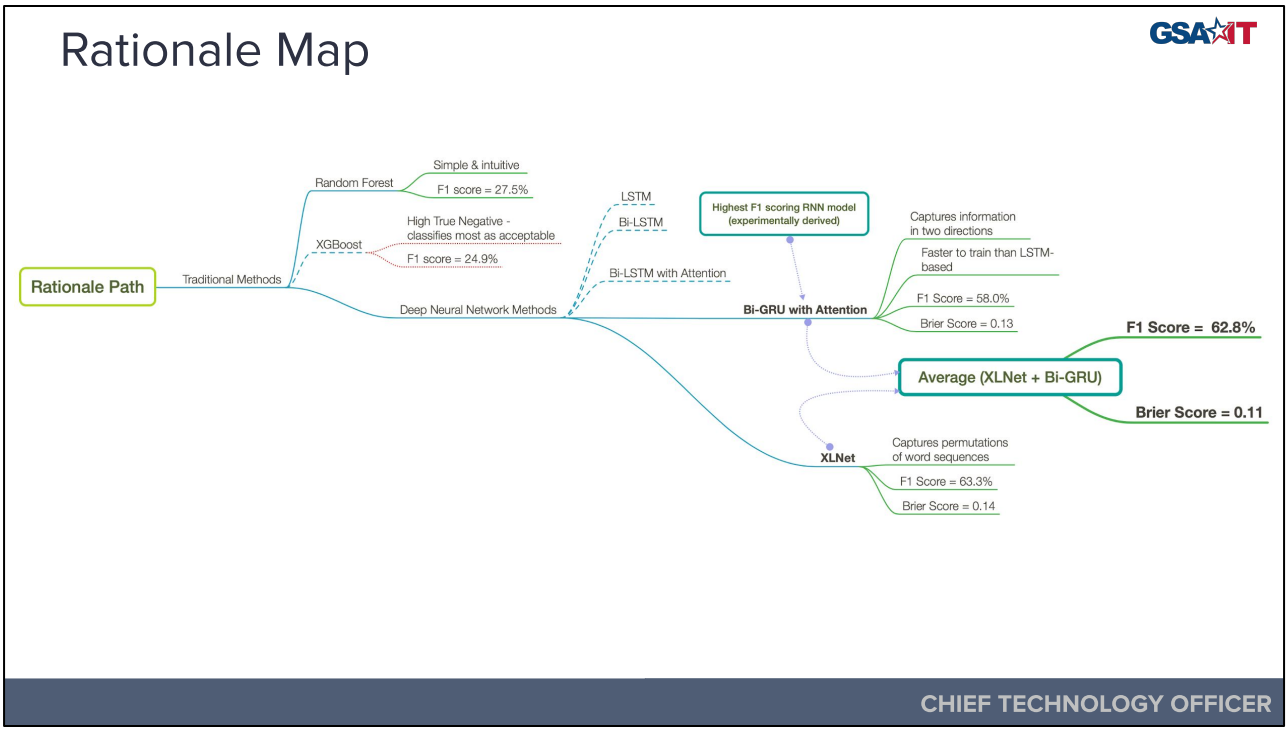


Figure 1: Overview of our solution presented as a flowchart.

This shows their finished architecture.



This is how the team described their process for deciding on the machine learning libraries to use:

- As a first step, our team used traditional algorithms to approach the problem.
- We started with **Random Forest**, which is a popular machine learning algorithm that is widely used in classification tasks.
- **Random Forest** is simple and intuitive in nature. It does not require hyperparameter tuning and usually does not overfit to the dataset with an increase in the number of decision trees within the model. We achieved an F1 score of 27.5% with it.
- As the next step, we implemented the **XGBoost** algorithm. XGBoost is a tree-based algorithm (like Random Forest) but uses the technique of boosting. Boosting is an error-correction algorithm which gives a higher emphasis on data-points which are misclassified. Unlike random forests, the decision trees are created iteratively, where at each step, the tree puts more emphasis on the misclassified points, so as to reduce the overall error. We therefore used XGBoost as the natural next step to Random Forest. On testing however, the F1 score achieved through XGBoost was lower than Random Forest. This was potentially due to a high False Negative, resulting in a low Recall value. This means that the model was classifying most clauses

- (even the ones that were labelled unacceptable) as acceptable.
- After running several experiments trying to improve the accuracy of traditional models, we realized that more advanced, deep learning based models could potentially help us increase accuracies. We therefore started with the simplest form of sequence models that are used on textual data: **Recurrent Neural Networks (RNN)**. As a starting point we used a **Long Short-Term Memory (LSTM)** which attempts to resolve the vanishing gradient, a known obstacle for RNNs. LSTMs however capture the flow of information in one direction (left to right in case of sentences). **Bidirectional LSTMs** capture the flow of information from both left to right and right to left. This serves as an advantage as the model can learn from the future and the past information at a given point in the sentence. To add interpretability to our results, we added an **attention** layer that provides the importance of words in the decision making process of the model. The attention layer weighs those words differently providing more emphasis on words that have stronger relationships. Experiments run by researchers and practitioners have shown that **Gated Recurrent Units (GRUs)** train faster and provide a proportional (or sometimes better as in our case) accuracies to that of LSTMs. We therefore replaced the LSTM units with GRUs. The GRU unit does not have a forget gate (unlike the LSTM) and has fewer parameters to train on.

- Arguably the most sophisticated classification model, as of now, is the **XLNet** which builds upon the transformer architecture which incorporates encoding and decoding layers in addition to sinusoidal position encoding of words in a sentence. It also incorporates permutations of words to learn more complex relationships between words. It has been pre-trained on a large corpus of data from Wikipedia, BookCorpus, etc. In our experiments, it did improve the validation accuracies.
- To provide an ensemble solution, we combined our best performing models: **XLNet and Bi-GRU with attention**. The final classification decision made by this ensemble was the average of the probabilities coming out of these models. We had lively discussions about whether we should just keep XLNet or do a weighted combination of XLNet with Bi-GRUs. The result was that the argument for robustness in the estimate ensembling these two complex algorithms won the argument of the day. In addition, in our final solution, we kept the Random Forest score as a sanity check output in our display. We believe this model is

- the easiest to explain to lay people. However this score was not used in the average.

Data Science Preprocessing and Modeling

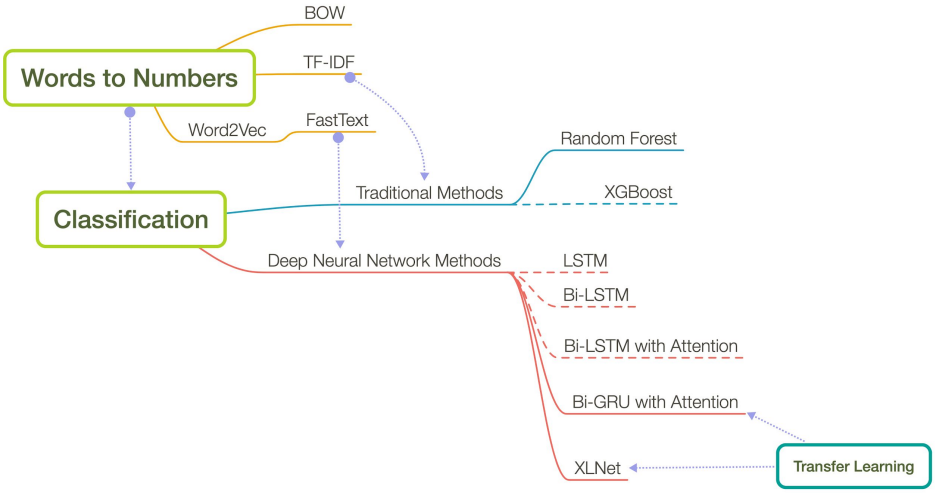


Figure 2: The expanded version of the back-end prediction engine.

From the team:
It comprises two major components: Words to Numbers and Classification. Arrows show which words to numbers algorithm used for which classification algorithm. Solid lines represent the models that were integrated into the final solution, while dashed lines represent the ones that we tested during the development process of our final solution.

Thanks!